Spy vs Spy: SELinux and MAC

David Collier-Brown

Not *that* Mac, Mandatory Access Control

- Controls that the machine owner sets to prohibit over-sharing of information
- Controls that the NSA wrote specifications for, because they needed them
  - But the machines were too expensive
  - And they didn't use crypto, what was even more expensive
  - And only worked on Mainframes
- So the NSA eventually gave up on them

Boy was *That* a Good Idea!

- Without MAC, a sysadmin has access to everything
- And of course, we won't leak anything.

- Unless, of course,
  - your company is arguably breaking the law
  - you're a crook
  - Both you and your company are crooks
  - Or you're a spy for the FSB (ie, KGB)

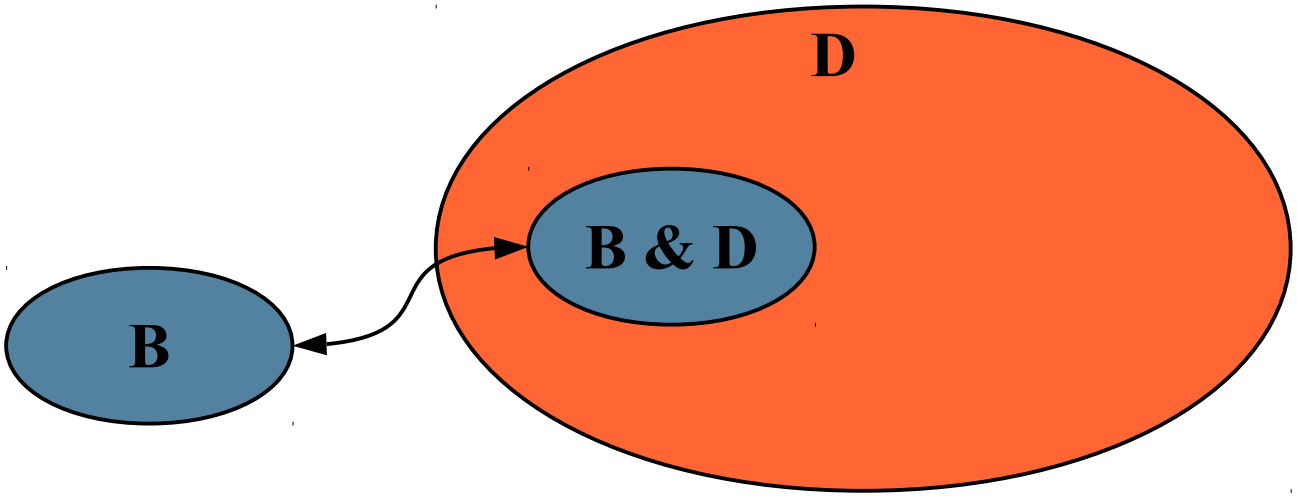
Mandatory Access Control is Ancient

- Invented in the Mainframe era
  - Ran on Multics, Trusted Solaris/HP
- Demanded by the U.S. Military
- Security in the sense of confidentiality
  - Does not address other attacks
  - Does keep person A out of person B's stuff
  - Helps prevent data escape once you've been hacked
- Turned *off* in SE Linux
  - Do we see a pattern developing here?

What it was Supposed to Do

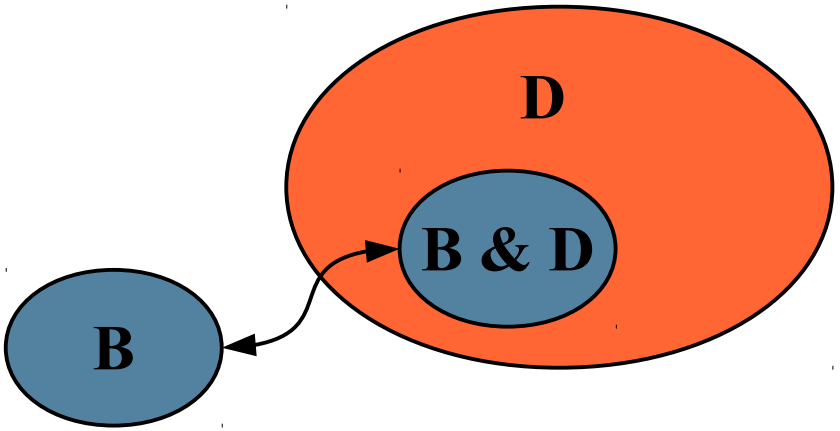
- Require two people to authorize a release
  - Like the “two signatures” rule in accounting
  - Forces thefts to be conspiracies
- Keep “top secret” away from people with just “confidential”
- Keep Bletchley Park's secrets separate from the Commando's stuff
  - Commandos can get captured by the enemy
  - You don't want all your secrets in one basket

For Civilians, Separation is the Big Thing



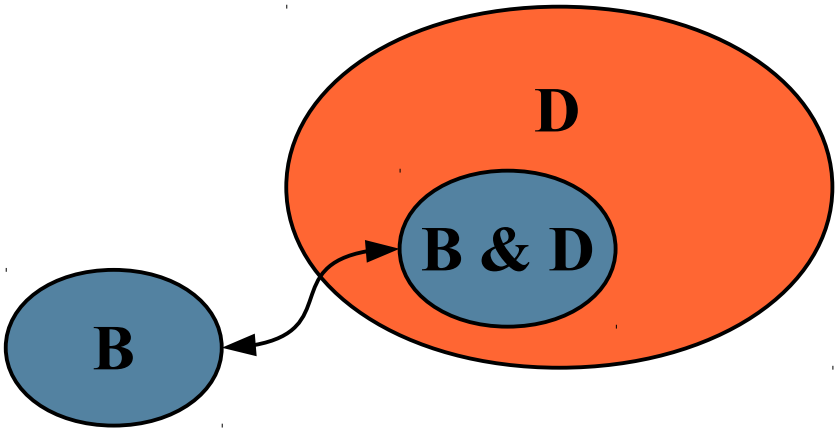
- I have a bank, who I trust (only with money)
- Should the bank app have access to anything on my phone?

How about Venn Diagrams?



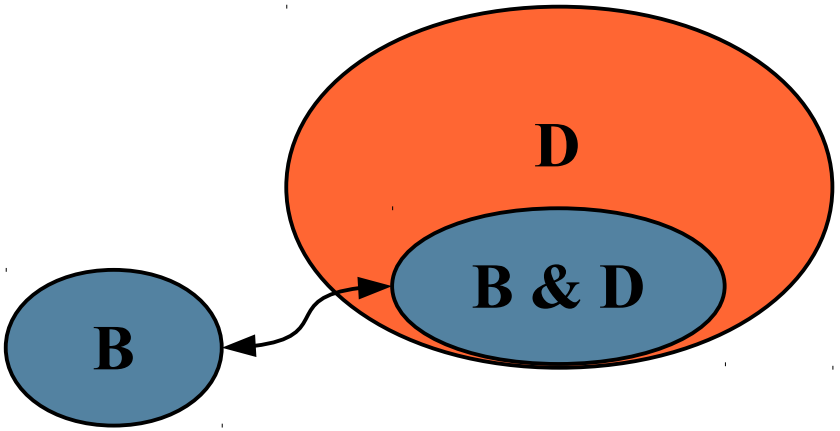
- Imagine I have labels that look like this
  - Implemented with private/public keys
- B(ank) can only see things labeled B
- D(ave) can only see things labeled D

Protecting the Bank from Dave



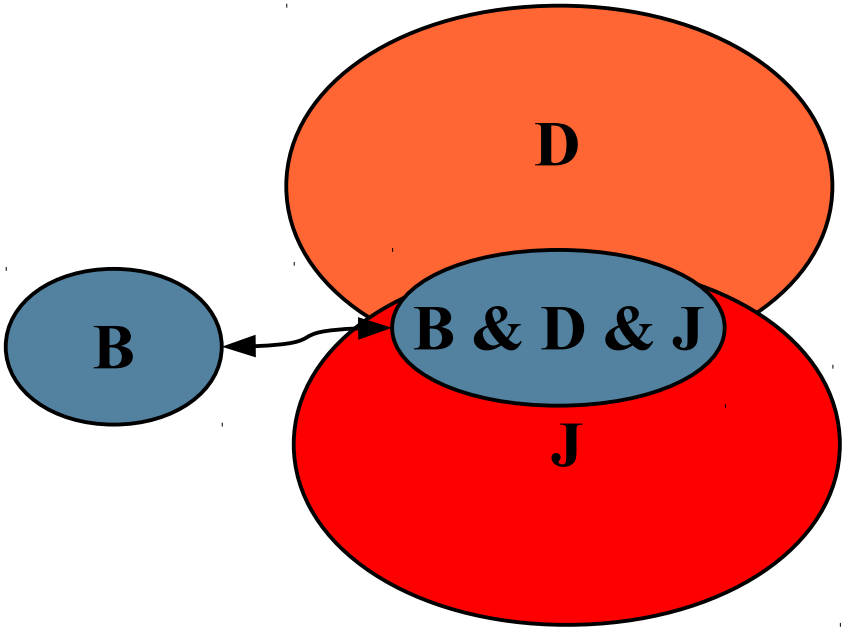
- The bank can communicate across untrusted links
- It can decide to only decrypt safe things, like printable statements
- Those end up labeled D, so I can print them

Protecting Dave From the Bank



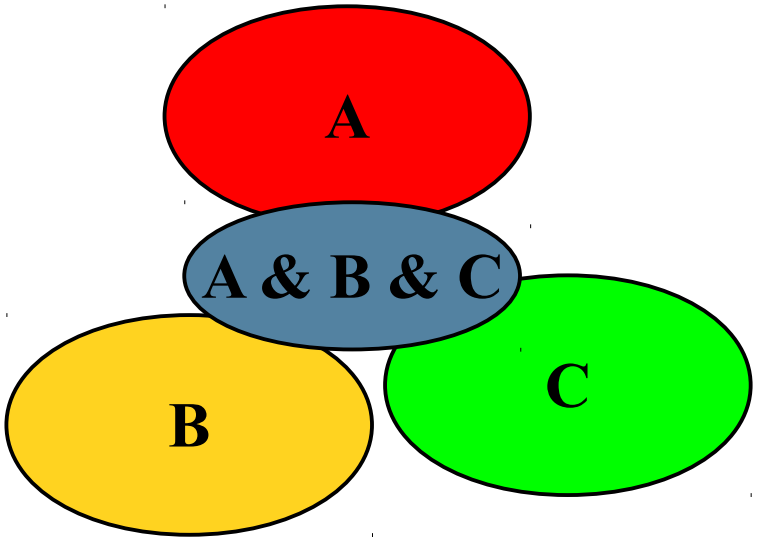
- The bank can't read anything except (B & D)
- If someone subverts the bank, they still have crack my security, not just steal all the “D” stuff
- Defense in depth

My Wife and I vs the Bank



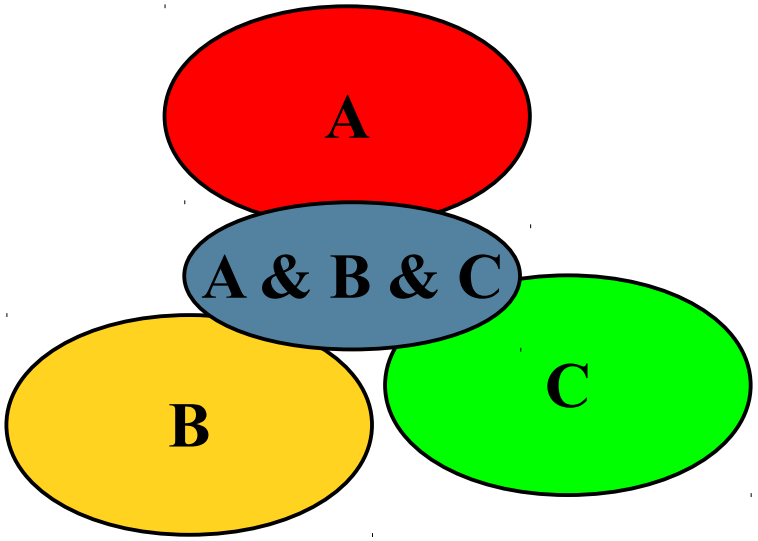
- We can also share stuff
- Some of our accounts belong to both me and my wife
- It takes agreement from all three
- We also have (B & (J | D)) accounts

My Company Would Like That, Too



- I work for company C
- We have customers A and B
  - A distrusts B
  - B distrusts A
- How do we cooperate?
- (A & B & C)

Let's Say I Want Something ...

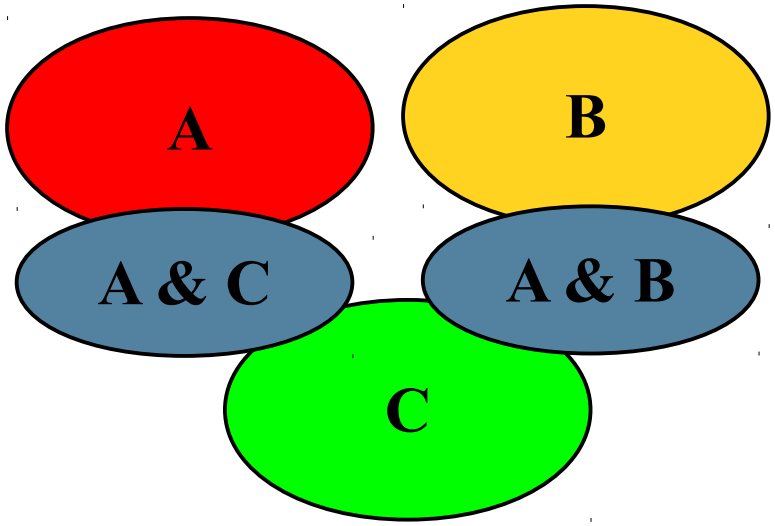


- I want to give part of the aircraft plans we're writing to someone else inside C
- I have to get
  - A to take his label off
  - B to take his label off
- Otherwise C can't read it

It's easier if we want to open-source it

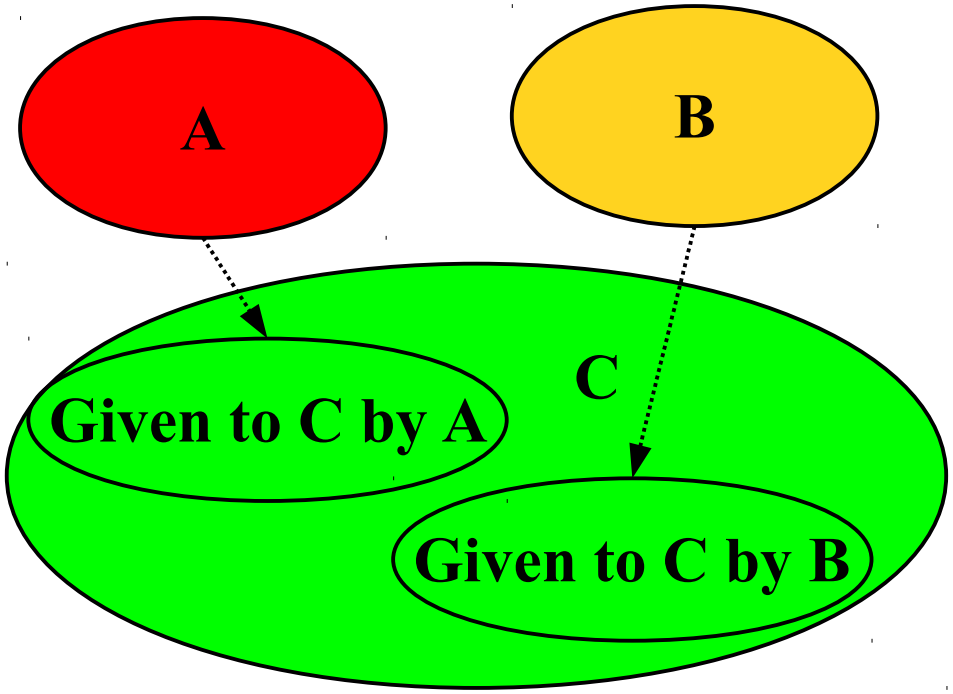
- We agree that we'll all take our labels off after 6 months
  - Historically, that was easier than getting A to trust B
- If anyone renigs, both other parties get mad and *want* to renegotiate

A and B Would Really Prefer



- A doesn't trust B
- Because C works with B, they don't really trust C all that much
- They try to create this with contracts, but ...

What They Really Get is This



- The contracts don't affect reality
- C gets trusted without limit
- And C is as trustworthy as … ***a sysadmin***

MAC scales

- The number of individual parties can be large
  - > We originally thought 36 was too many
  - > But that was on a mainframe
- We went to 72 with a declaration change
  - > The code didn't care, and it ran just as well
- In principle it can be be large as you like
- Banks will want 2^128 or so

What Doesn't Scale

- Keeping track of keys
  - Humans do this badly
- Companies do it well
  - Every company I deal with has a customer number that's me
  - My problem is keeping them from comparing notes
- These days I have a “Password Manager”
- *Real Soon Now* I'll have a “key manager”
  - I already have a keyring

Complex Stuff Doesn't Scale

- Top secret, secret, confidential, restricted, unclassified:
  - 5 layers only an army could love.
  - And maybe add “unclassified but sensitive”
  - And “system high” and “system low”
- The rulesets are O(N!) hard, where N >= 6.
- I'd like N = 1

Really Subtle Stuff Doesn't Scale

- The “Simple Security” property
  - allows a person with “top secret” to read “secret” files, which makes some sense
- The “*” Property
  - A person with secret can write top secret files
  - We actually used that for un-erasable logfiles
- Between them, the complexity makes mathematicans cry bitter tears

Spy versus Spy

- A really serious spy is going to get me
  - If the director of the FBI wants to know my sexual preference, he's going to find out
- Right now a script kiddie can eat my lunch
  - My phone is an attractive nuisance
  - But that same phone has more power than HI-Multics.ARPA
- I want to “raise the bar”
- I want it to take J. Edgar Hoover to hack me

I used to run Trusted Solaris

- I had the “company C” problem
  - I had secure links to A and B
  - Neither A nor B trusted us
- So my boss send me on a week course just to learn how to sysadmin Trusted Solaris
  - It made my brain hurt
  - It was also pretty impressive
- The standard it met was set by the NSA
  - Remember I said they didn't use it?

The NSA is Stupid

- Ok, they're brilliant
  - and stupid at the same time
- The shoemaker's children go barefoot
  - They designed systems to keep spies out
  - They then arguably weakened them so others wouldn't benefit
  - And argued that weaker was all anyone needed
  - Then they took their own advice, and decided that they didn't need confidentiality either

SE Linux is brilliant

- Ok, and stupid at the same time
- It had the basic structures
  - It has the labels
  - It has MAC turned off
- It's being used to address other kinds of attacks, notably privilege escalation
- But it doesn't use crypto for labeling
  - Despite being on an insanely fast machine

Other Advances We Already Have

- Ensuring integrity of embedded devices like phones, to reject bad programs – https://lwn.net/Articles/568943/
- Controlling access to one's data (a key store for individuals) - http://www.newscientist.com/article/mg22029374.600-private-data-gatekeeper-stands-between-you-and-the-nsa.html
- End to end encryption – silent circle: https://silentcircle.com/

What I Want

- HI-Multics.ARPA on my wristwatch
- Military grade security, without any of the complex stuff
- Controlled sharing without having to trust everyone's sysadmin
- Commercial key stores in multiple countries, each with 1/10 of each of my keys
- The NSA back spying on the FSB
  - because it's too much work for too little payoff to bother spying on me